



H-1141

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

K. Serizawa et al.

ATTN: Manager,
Applications Branch

Serial No. 10/825,158

Filed: April 16, 2004

For: METHOD FOR ALLOCATING STORAGE AREA
TO VIRTUAL VOLUME

TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith is a certified priority document
(JP 2003-150082) of a corresponding Japanese patent
application for the purpose of claiming foreign priority under
35 U.S.C. § 119. An indication that this document has been
safely received would be appreciated.

Respectfully submitted,

Daniel J. Stanger
Registration No. 32,846
Attorney for Applicants

MATTINGLY, STANGER & MALUR
1800 Diagonal Road, Suite 370
Alexandria, Virginia 22314
(703) 684-1120
Date: August 16, 2004

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
る事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出 願 年 月 日 2 0 0 3 年 5 月 2 8 日
Date of Application:

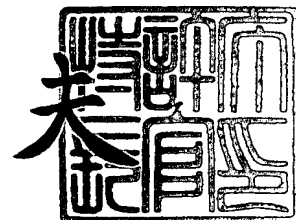
出 願 番 号 特 願 2 0 0 3 - 1 5 0 0 8 2
Application Number:
ST. 10/C]: [J P 2 0 0 3 - 1 5 0 0 8 2]

願 人 株式会社日立製作所
Applicant(s):

2 0 0 3 年 1 1 月 2 6 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康 夫



【書類名】 特許願

【整理番号】 K03001961A

【あて先】 特許庁長官殿

【国際特許分類】 G06F 12/00

【発明者】

 【住所又は居所】 神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所システム開発研究所内

 【氏名】 芹沢 一

【発明者】

 【住所又は居所】 神奈川県横浜市戸塚区戸塚町 5 0 3 0 番地 株式会社日立製作所ソフトウェア事業部内

 【氏名】 森田 眞司

【発明者】

 【住所又は居所】 神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所システム開発研究所内

 【氏名】 岩見 直子

【特許出願人】

 【識別番号】 000005108

 【氏名又は名称】 株式会社 日立製作所

【代理人】

 【識別番号】 100075096

 【弁理士】

 【氏名又は名称】 作田 康夫

【手数料の表示】

 【予納台帳番号】 013088

 【納付金額】 21,000円

【提出物件の目録】

 【物件名】 明細書 1

 【物件名】 図面 1

【物件名】 要約書 1
【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 記憶領域割当方法、システム及び仮想化装置

【特許請求の範囲】

【請求項 1】

計算機、前記計算機に接続される仮想化装置及び前記仮想化装置に接続される記憶装置において前記計算機に前記記憶装置が有する記憶領域を割当てて方法であって、

仮想化装置で予め前記計算機に仮想的な記憶領域を割当て、

前記計算機が前記仮想的な記憶領域を使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持し、前記情報を元に、前記計算機に割当てられた前記仮想的な記憶領域に前記記憶装置が有する記憶領域を割当ててことを特徴とする記憶領域割当方法。

【請求項 2】

請求項 1 記載の記憶領域割当方法であって、

前記仮想的な記憶領域に前記計算機がデータを書き込む時に、前記記憶装置が有する記憶領域から前記仮想的な記憶領域に使用されていない記憶領域を選択し、前記仮想的な記憶領域中の前記書き込みの対象となる記憶領域に前記選択された記憶領域を割当ててことを特徴とする記憶領域割当方法。

【請求項 3】

請求項 2 記載の記憶領域割当方法であって、

前記仮想的な記憶領域に割当てられた前記記憶装置の有する記憶領域が所定の条件を満たす場合に、前記情報を更新し、前記仮想的な記憶領域に対する前記記憶装置の有する記憶領域の割当を止めることを特徴とする記憶領域割当方法。

【請求項 4】

請求項 3 記載の記憶領域割当方法であって、

前記所定の条件とは、前記計算機が、前記仮想的な記憶領域に割当てられた前記記憶装置が有する記憶領域を使用しなくなったという条件であることを特徴とする記憶領域割当方法。

【請求項 5】

請求項 1 記載の記憶領域割当方法であって、

前記計算機からの所定の書き込みである場合には、前記仮想的な記憶領域に割当て前記記憶装置が有する記憶領域の容量を他の書き込みで割当てられる前記記憶装置が有する記憶領域の記憶容量と異ならせることを特徴とする記憶領域割当方法。

【請求項 6】

請求項 5 記載の記憶領域割当方法であって、前記所定の書き込みとは、前記計算機によるファイルシステムの初期化处理の際に発生する書き込みであることを特徴とする記憶領域割当方法。

【請求項 7】

請求項 6 記載の記憶領域割当方法であって、前記ファイルシステムがジャーナル領域を使用する場合に、前記ジャーナル領域へ前記計算機が書き込んだ情報に対応したファイルシステム上のファイルの消去または縮小を特定し、この消去または縮小によって、ファイルシステム上のどのデータも割当てられなくなった前記仮想的な記憶領域に対応する前記記憶装置が有する記憶領域を、前記仮想的な記憶領域への割当から外すことを特徴とする記憶領域割当方法。

【請求項 8】

計算機及び記憶装置と接続される仮想化装置であって、

前記計算機と接続されるポートと、

前記記憶装置と接続されるポートと、

前記ポートから入力されるデータを他の前記ポートに転送する転送部と、

制御部と、

記憶部とを有し、

前記記憶部は、前記計算機に提供される仮想的な記憶領域と前記記憶装置が有する記憶領域との対応関係についての情報を保持し、

前記制御部は、前記ポートを介して前記計算機から前記仮想的な記憶領域に対する書き込み要求を受け付け、前記書き込み要求に対応する前記仮想的な記憶領域に前記記憶装置が有する記憶領域が割当てられているか否かを前記情報に基づいて判断し、前記仮想的な記憶領域に前記記憶装置が有する記憶領域が割当てら

れていない場合、未だ前記仮想的な記憶領域に割当てられていない前記記憶装置が有する記憶領域を選択し、前記仮想的な記憶領域に前記選択された前記記憶装置が有する記憶領域を割当て、前記情報の内容を更新することを特徴とする仮想化装置。

【請求項 9】

請求項 8 記載の仮想化装置であって、前記制御部は、前記仮想的な記憶領域が前記計算機に使用されなくなった場合に、前記情報を更新して、使用されなくなった前記仮想的な記憶領域に割当てられていた前記記憶装置が有する記憶領域の割当を解消することを特徴とする仮想化装置。

【請求項 1 0】

請求項 9 記載の仮想化装置であって、前記制御部は、前記計算機からの書き込み要求の内容に応じて、前記仮想的な記憶領域に割当てる前記記憶装置が有する記憶領域の容量を決定することを特徴とする仮想化装置。

【請求項 1 1】

計算機及び記憶装置と接続される仮想化装置が有する記憶部に格納され、前記仮想化装置が有する制御部で実行されるプログラムであって、

予め前記計算機に仮想的な記憶領域を割当てるモジュールと、

前記計算機が前記仮想的な記憶領域を実際に使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持するモジュールと、

前記情報を元に、前記計算機に提供される前記仮想的な記憶領域に前記記憶装置が有する前記記憶領域を割当てるモジュールとを有することを特徴とするプログラム。

【請求項 1 2】

記憶装置と、

計算機と、

前記記憶装置及び前記計算機と接続される仮想化装置とを有し、

前記仮想化装置は、

予め前記計算機に仮想的な記憶領域を割当て、

前記計算機が前記仮想的な記憶領域を使用する際に、前記仮想的な記憶領域と前記記憶装置が有する記憶領域とを対応付けてその情報を保持し、前記情報を元に、前記計算機に提供される前記仮想的な記憶領域に前記記憶装置が有する記憶領域を割当ててることを特徴とするシステム。

【請求項 13】

請求項 12 記載のシステムであって、

更に前記仮想化装置に接続される管理用の計算機を有し、

前記仮想化装置は、前記管理用の計算機から入力される情報に基づいて前記仮想的な記憶領域を前記計算機に割当ててることを特徴とするシステム。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、記憶装置システムが有する記憶領域の計算機への割当て技術、特に、仮想的な記憶装置システムにおける記憶領域の割当て技術に関する。

【0002】**【従来の技術】**

複数の記憶装置又は記憶装置システムを仮想化して一つの大規模な記憶装置システムとして計算機等に提供する装置が、特許文献 1 に開示されている。特許文献 1 によれば、装置において計算機等に提供される仮想的な記憶装置システムの仮想化の設定を変更することで、計算機に提供する仮想的な記憶領域のサイズを拡大することが出来る。

【0003】

また、特許文献 1 における仮想的な記憶領域の拡大を計算機側で認識するためのソフトウェア技術に関しては、非特許文献 1 に開示されている。

【0004】**【特許文献 1】**

特願 2002-330323 号公報

【非特許文献 1】

LVM HOWTO (<http://www.linux.org/docs/>

ldp/howto/LVM-HOWTO-9.htmlから入手)の 9.9
Extending a logical volume

【0005】

【発明が解決しようとする課題】

上述したように、特許文献1に示す仮想的な記憶領域のサイズを拡大する場合、非特許文献1に示すような計算機で使用されるソフトウェアが必須となる。これは、計算機が使用している記憶領域のサイズを動的に拡大することが可能である記憶装置又は記憶装置システムが従来存在せず、計算機がそのような記憶領域のサイズ拡大を想定していないためである。

【0006】

しかし、このようなソフトウェアが必要になることは、以下のような管理コストの増大を引き起こす。即ち、計算機が複数ある場合に、その全てにソフトウェアを追加する必要があること、計算機の種類が複数ある場合それぞれに対応し相異なるソフトウェアを追加する必要があること、さらに、計算機で使用されるオペレーティングシステムがバージョンアップした場合に、それに追従して、対応するソフトウェアを更新する必要があるからである。

【0007】

ところで、計算機が使用している記憶領域のサイズが拡大される理由は、最初から大きなサイズの記憶領域を計算機に割当てると、割当てられた記憶領域に未使用領域が発生して記憶領域に無駄が生じるので、計算機が使用するデータ量の増加に応じて記憶領域を割当てようとするからである。

【0008】

したがって、計算機には最初から大きな記憶領域を割当てつつ、実際には割当てられた記憶領域の未使用領域が発生させないようにすれば、上述のような仮想的な記憶領域のサイズの拡大とそれに伴うソフトウェアの導入を行う必要がない。

【0009】

つまり本発明の目的は、計算機に対して効率的に記憶領域を割当てることが可能な仮想化された記憶装置システム及び仮想化された記憶装置システムを計算機

に提供する装置を提供することである。

【0010】

【課題を解決するための手段】

本発明は、計算機、計算機と接続され、仮想化された記憶領域を接続された計算機に提供する装置（以下「仮想化装置」）及び装置に接続される記憶装置とを有する構成とする。そして、仮想化装置は、計算機に提供する仮想化された記憶領域を作成する際に、記憶装置が有する記憶領域を実際には仮想化された記憶領域には割当てないが、計算機には割当てたように通知する。その後、計算機が仮想化された記憶領域へデータの書き込みを行ったときに初めて、仮想化された記憶領域中のデータの書き込みに対応する記憶領域に記憶装置の記憶領域を割当てる。

【0011】

更に、仮想化装置は、仮想化された記憶領域に割当てられた記憶装置の記憶領域が計算機に使用されなくなった場合に、その記憶装置の記憶領域と仮想化された記憶領域との対応関係の情報を消去することでその記憶装置の記憶領域を仮想化された記憶領域から解放する。

【0012】

尚、仮想化された記憶領域に割当てられる記憶装置の記憶領域の容量は、全て同じでも良く、割当てごとに異なる容量とする実施形態も考えられる。

【0013】

【発明の実施の形態】

以下、本発明の実施形態を、図面を用いて説明する。

【0014】

図1は、本発明を適用したシステムの第一の実施形態の全体構成を示す図である。

【0015】

システムは、少なくとも1台のホストプロセッサ12、少なくとも1台のストレージ装置13、仮想化スイッチ11及び管理コンソール14を有する。

【0016】

ホストプロセッサ 12 は、ストレージ装置 13 に格納されたデータを使用する計算機である。ホストプロセッサ 12 は、仮想化スイッチ 11 が提供する記憶領域を仮想化スイッチ 11 に接続されていない他の計算機に提供する機能を有するファイルサーバでも良い。

【0017】

ストレージ装置 13 は記憶装置又は記憶装置システムである。ここで、記憶装置とは、ハードディスクドライブやDVDドライブ等の単体の記憶装置を指し、記憶装置システムとは、制御部及びハードディスクドライブ等のディスク装置を有するストレージサブシステムを指すものとする。また、ストレージ装置 13 は、少なくとも 1 つの論理ユニット（以下「LU」）131 を有する。LU 131 は、ストレージ装置 13 が有する物理的な記憶領域から構成されている論理的な記憶領域である。LU 131 は、ホストプロセッサ 12 などストレージ装置 13 に接続される装置には、論理的に独立した 1 つのストレージ装置として認識される。

【0018】

また、LU 131 は、複数の部分的な論理的記憶領域（以下「実領域」）132 から構成される。実領域 132 の各々も、ストレージ装置 13 が有する物理的な記憶領域に対応する。実領域 132 のサイズは任意であり、その範囲は、連続したアドレスをもつ領域である。

【0019】

仮想化スイッチ 11 は、図示するように通信線やスイッチによって他の装置と接続されており、他の装置と通信することができる装置である。また、仮想化スイッチ 11 は、仮想化スイッチ 11 自身に接続されている一つ又は複数のストレージ装置 13 が有する記憶領域をまとめて一つまたは複数の記憶領域とする（以下「仮想化」と称する）仮想化装置である。そして、仮想化スイッチ 11 は、仮想化した記憶領域を、仮想化スイッチ 11 に接続されるホストプロセッサ 12 等に提供する。

【0020】

仮想化スイッチ 11 とホストプロセッサ 12 との間および仮想化スイッチ 11

とストレージ装置 13 との間で使用される通信線やスイッチでは、ファイバチャネル等のプロトコルが使用される。ただし、使用される通信線やプロトコルはローカルエリアネットワーク等で使用される通信線やプロトコルでも良い。仮想化スイッチ 11 は、ホストプロセッサ 12 とストレージ装置 13 との間に接続され、ホストプロセッサ 12 が発行するコマンドをストレージ装置 13 側に転送する機能を有する。また、仮想化スイッチ 11 がホストプロセッサ 12 に対して提供する仮想的な記憶領域を以下仮想ボリューム 100 と称する。

【0021】

仮想ボリューム 100 は、少なくとも 1 つの実領域 132 から構成される仮想化された記憶領域である。仮想化スイッチ 11 は、複数の仮想ボリューム 100 をホストプロセッサ 12 等に提供することが出来る。仮想ボリューム 100 の各々には、仮想ボリュームを特定するための仮想化スイッチ 11 内で一意の識別子（以下「仮想ボリューム識別子」）が与えられている。また、個々の仮想ボリューム 100 の記憶領域には連続したアドレスが付されている。ホストプロセッサ 12 は、ストレージ装置 13 の LU 131 内の実領域 132 を直接指定する代わりに、仮想ボリューム識別子及び仮想ボリューム 100 内の場所を示すアドレスを指定して、ストレージ装置 13 に格納されたデータを利用する。

【0022】

管理コンソール 14 は、仮想ボリューム 100 を作成するためにシステム管理者によって使用される計算機であり、表示装置と入力装置を備える。管理コンソール 14 は、仮想化スイッチ 11 とネットワークを介して接続されている。

【0023】

図 2 は、仮想化スイッチ 11 の内部構成を示す図である。

【0024】

仮想化スイッチ 11 は、入力ポート 240、出力ポート 250、転送部 230、制御部 210 及び記憶部 220 を有する。入力ポート 240 は、仮想化スイッチ 11 がホストプロセッサ 12 と通信するための通信線と接続されるポートである。出力ポート 250 は、仮想化スイッチ 11 がストレージ装置 13 と通信するための通信線と接続されるポートである。尚、入力ポート 240 及び出力ポート

250を構成する装置は同一であっても良い。この場合、どのポートを入力ポートあるいは出力ポートとして使用するかは、使用者が選択する。

【0025】

転送部230はメモリを有し、そのメモリに転送情報テーブル231を保持する。転送情報テーブル231には、各入力ポート240を介して仮想化スイッチ11と通信可能なホストプロセッサ12及び各出力ポート250を介して仮想化スイッチ11と通信可能なストレージ装置13との間の対応関係についての情報が格納される。

【0026】

転送部230は、転送情報テーブル231を参照し、入力ポート240がホストプロセッサ12から受信した入出力要求を、要求先のストレージ装置13と仮想化スイッチ11との間の通信に使用される出力ポート250へ転送する。また、転送部230は、出力ポート250がストレージ装置13から受信した応答情報やデータを、データ等を受信すべきホストプロセッサ12と仮想化スイッチ11の間の通信に使用される入力ポート240へ転送する。ただし、ホストプロセッサ12から受け取った入出力要求が仮想ボリューム100に対する入出力要求である場合には、転送部230は、後述のアクセス変換プログラム212に基づく制御部210の処理において選択されたストレージ装置13へ入出力要求を送信する。

【0027】

制御部210はプロセッサ及びメモリを有し、そのメモリに仮想ボリューム定義プログラム211、アクセス変換プログラム212及び割当処理プログラム213の各プログラムが格納される。これらのプログラムは、制御部210のプロセッサで実行される。

【0028】

記憶部220は、仮想ボリューム管理テーブル221、実領域管理テーブル222及びアクセス変換テーブル224を記憶する。

【0029】

アクセス変換テーブル224は、仮想化スイッチ11がホストプロセッサ12

に提供する仮想ボリューム 1 0 0 ごとに存在する。アクセス変換テーブル 2 2 4 は、エントリ 3 3 1 と、対応する仮想ボリューム 1 0 0 の仮想ボリューム識別子を登録するエントリ 3 3 2 とを保持する。個々のエントリ 3 3 1 には、仮想ボリューム 1 0 0 内の記憶領域を示すアドレス範囲と、そのアドレス範囲に対応する実領域 1 3 2 が属する L U 1 3 1 を指定する識別子である L U アドレス及び実領域 1 3 2 の L U 1 3 1 内における位置を示す L U 内アドレスとの対応関係の情報が登録される。即ちアクセス変換テーブル 2 2 4 は、仮想ボリューム 1 0 0 の記憶領域のアドレスとストレージ装置 1 3 の記憶領域のアドレスとの対応情報を保持している。

【 0 0 3 0 】

なお、仮想ボリューム 1 0 0 に実領域 1 3 2 が割当てられていない場合、エントリ 3 3 1 の L U アドレス及び L U 内アドレスの情報が登録される部分には実領域 1 3 2 が割当てられていないことを示す情報、具体的には -1 が登録される。アクセス変換テーブル 2 2 4 は、仮想ボリューム 1 0 0 の記憶領域の構成が変更されたとき、制御部 2 1 0 によって更新される。

【 0 0 3 1 】

制御部 2 1 0 は、アクセス変換プログラム 2 1 2 を実行し、アクセス変換テーブル 2 2 4 を参照して、ホストプロセッサ 1 2 から受け取った仮想ボリューム 1 0 0 に対する入出力要求を、対応するストレージ装置 1 3 が有する L U 1 3 1 への入出力要求に変換する。また、対応する L U 1 3 1 が存在しない場合、制御部 2 1 0 は、仮想ボリューム定義プログラム 2 1 1 を実行して、仮想ボリューム 1 0 0 の定義変更処理を実行する。なお、性能向上のために、仮想化スイッチ 1 1 は、アクセス変換テーブル 2 2 4 およびアクセス変換プログラム 2 1 2 を各入力ポート 2 4 0 ごとに有していても良い。

【 0 0 3 2 】

実領域管理テーブル 2 2 2 は、L U 1 3 1 ごとに存在する。実領域管理テーブル 2 2 2 は、L U 1 3 1 に含まれる実領域 1 3 2 を管理するために使用されるテーブルである。各実領域管理テーブル 2 2 2 には、ストレージ装置 I D、L U アドレス及び実領域リスト 3 2 4 が格納される。ストレージ装置 I D は、L U 1 3

1 を保持するストレージ装置 13 を示す識別子である。

【0033】

実領域リスト 324 は、少なくとも 1 つのエントリ 325 を有する。各エントリ 325 は、LU131 を構成する個々の実領域 132 に対応して設けられ、実領域 ID、サイズ及び仮想ボリューム識別子の情報を登録する項目を有する。実領域 ID はエントリ 325 に対応する実領域 132 を特定するための識別子、サイズは実領域 132 のサイズである。またエントリ 325 に登録される仮想ボリューム識別子は、実領域 132 が割当てられている仮想ボリューム 100 に割り振られた仮想ボリューム識別子である。実領域リスト 324 内のエントリ 325 は、実領域 132 のアドレス順に配列されている。

【0034】

なお、本実施形態では、実領域 132 のサイズは固定であると仮定するため、エントリ 325 に実領域 132 のサイズについての情報を登録する項目を含めなくても良い。また、未使用である実領域 132 に対応するエントリ 325 の仮想ボリューム識別子を登録する項目には、未使用であることを示す null が登録される。

【0035】

このように、実領域管理テーブル 222 は、LU131 に属する各実領域 132 が仮想ボリューム 100 として利用されているか否かをに関する情報を保持しており、仮想化スイッチ 11 が仮想化ボリューム 100 に新たに割当て実領域 132 を選択する際に使用される。

【0036】

尚、実領域管理テーブル 222 は、仮想化スイッチ 11 にストレージ装置 13 が接続された場合等のタイミングで、管理端末 14 を介した管理者の指示に基づいて作成される。さらに、この際に、LU131 とそれを構成する実領域 132 の記憶容量等が決定される。尚、実領域管理テーブル 222 が作成された時点では、全てのエントリ 235 の領域 ID には仮想化スイッチ 11 内で一意な識別子がそれぞれ書き込まれ、仮想ボリューム識別子には null が設定される。

【0037】

仮想ボリューム管理テーブル 2 2 1 は、仮想ボリューム 1 0 0 ごとに存在する。各仮想ボリューム管理テーブル 2 2 1 には、識別子エントリ 3 1 1 及び実領域リスト 3 1 5 が格納される。識別子エントリ 3 1 1 には、テーブル 2 2 1 に対応する仮想ボリューム 1 0 0 の仮想ボリューム識別子が登録される。実領域リスト 3 1 5 は、テーブル 2 2 1 に対応する仮想ボリューム 1 0 0 にどの実領域 1 3 2 が割当てられているかを示すリストである。実領域リスト 3 1 5 の各エントリ 3 1 7 は、仮想ボリューム 1 0 0 上のアドレス順に並んでおり、そのアドレスに対応する実領域 1 3 2 の実領域 ID を格納している。当該仮想ボリューム 1 0 0 のうち、実領域 1 3 2 が割当てられていない部分に相当する実領域リスト 3 1 5 のエントリ 3 1 7 には、有効な実領域 ID 3 1 7 の代わりに空を示す null 値が格納される。

【 0 0 3 8 】

このように、仮想ボリューム管理テーブル 2 2 1 は、仮想ボリューム 1 0 0 の記憶領域がどの実領域 1 3 2 と対応付けられているかを示す情報を保持しており、仮想化スイッチ 1 1 が解放可能な実領域 1 3 2 を選択するために使用される。

【 0 0 3 9 】

以下、本実施形態における仮想化スイッチ 1 1 が行う記憶領域の割当て処理について説明する。

【 0 0 4 0 】

制御部 2 1 0 は、仮想ボリューム定義プログラム 2 1 1 を実行することで、仮想ボリューム 1 0 0 の定義を作成または変更する。制御部 2 1 0 は、管理コンソール 1 4 を介してシステム管理者から仮想ボリューム 1 0 0 の作成要求を受け取り、仮想ボリューム管理テーブル 2 2 1 及びアクセス変換テーブル 2 2 4 を新規作成する。この場合、制御部 2 1 0 は、識別子エントリ 3 1 1 に、既に作成された別の仮想ボリューム 1 0 0 と重複しない仮想ボリューム識別子を生成して格納し、実領域リスト 3 1 5 に空の値を設定することで実領域リスト 3 1 5 を初期化する。このように、仮想ボリューム 1 0 0 の生成直後は、仮想ボリューム 1 0 0 に実領域 1 3 2 が対応付けられない。従って、このとき制御部 2 1 0 は、新規作成される仮想ボリューム 1 0 0 に対応するアクセス変換テーブル 2 2 4 にも空の

値を登録し、アクセス変換テーブル 2 2 4 を初期設定する。

【0 0 4 1】

このように初期設定することで、仮想ボリューム 1 0 0 の作成時点では、ホストプロセッサ 1 2 には所定の大きさの仮想ボリューム 1 0 0 が割当てられたことを示す情報が通知されるが、その仮想ボリューム 1 0 0 にはストレージ装置 1 3 が提供する論理的な記憶領域である実領域 1 3 2 が割当てられていない状態となる。仮想化スイッチ 1 1 は、その後、ホストプロセッサ 1 2 等からのデータの書き込み要求を受信したタイミングで、仮想ボリューム 1 0 0 に実領域 1 3 2 を割当てる。このようにすることで、計算機に割当てる記憶領域の無駄を省くことが出来る。

【0 0 4 2】

したがって、実領域 1 3 2 を仮想ボリューム 1 0 0 に割当てるため、制御部 2 1 0 は、ホストプロセッサ 1 2 のデータ書き込み要求等の要求に従い、仮想ボリューム管理テーブル 2 2 1 を変更する。この場合、制御部 2 1 0 は、仮想ボリューム 1 0 0 に実領域 1 3 2 を割当てた後、アクセス変換テーブル 2 2 4 を更新する。尚、実際には、入力ポート 2 4 0 から受信した入出力要求の宛先が仮想ボリューム 1 0 0 であった場合に、転送部 2 3 0 は、その入出力要求を制御部 2 1 0 に転送する。制御部 2 1 0 は、転送された入出力要求について、以下の処理を実行する。また、図 3 の処理終了後、制御部 2 1 0 は変換した入出力要求の要求先の情報を転送部 2 3 0 に送信し、転送部 2 3 0 はその情報に基づいて入出力要求を各ストレージ装置 1 3 に転送する。

【0 0 4 3】

図 3 は、制御部 2 1 0 がホストプロセッサ 1 2 から入出力要求を受取った際に行う処理の手順を示すフローチャートである。

【0 0 4 4】

まず、制御部 2 1 0 は、ホストプロセッサ 1 2 から受け取った仮想ボリューム 1 0 0 に対する入出力要求が書き込み要求であるかどうかを判断する（ステップ 2 0 0 1）。

【0 0 4 5】

ホストプロセッサ 1 2 からの入出力要求が書き込み要求である場合、制御部 2 1 0 は、書き込み要求で指定された仮想ボリューム 1 0 0 のアドレスに実領域 1 3 2 が対応付けられているかどうかを、アクセス変換テーブル 2 2 4 で確認する（ステップ 2 0 0 2）。指定された仮想ボリューム 1 0 0 のアドレスに対応する実領域 1 3 2 がアクセス変換テーブル 2 2 4 に登録されていない場合、制御部 2 1 0 は、仮想ボリューム定義プログラム 2 1 1 を実行して仮想ボリューム 1 0 0 の定義変更処理を行う。具体的には、制御部 2 1 0 は、指定された仮想ボリュームのアドレスに実領域 1 3 2 を割当て。具体的には、実領域管理テーブル 2 2 2 から未使用の実領域 1 3 2 を検索して割当て（即ち実領域管理テーブル 2 2 2 および仮想ボリューム管理テーブル 2 2 1 を更新する）、アクセス変換テーブル 2 2 4 を更新する。尚、検索された実領域 1 3 2 の記憶容量が仮想ボリューム 1 0 0 のアドレス領域に対して不足する場合、制御部 2 1 0 は更に空きの実領域 1 3 2 を検索し、充当するまで検索を行う（ステップ 2 0 0 6）。

【 0 0 4 6 】

ステップ 2 0 0 6 の処理の後、ステップ 2 0 0 2 で実領域 1 3 2 が登録されていると判断された場合及びステップ 2 0 0 1 で書き込み要求ではないと判断された場合は、制御部 2 1 0 は、アクセス変換テーブル 2 2 4 を参照して、ホストプロセッサ 1 2 から受け取った仮想ボリューム 1 0 0 に対する入出力要求を対応するストレージ装置 1 3 の L U 1 3 1 への入出力要求に変換して（ステップ 2 0 0 9）、処理を完了する。

【 0 0 4 7 】

このようにステップ 2 0 0 6 の処理によって、仮想ボリューム 1 0 0 へデータが書き込まれる毎に、対応する実領域 1 3 2 を割当ててを可能にしている。

【 0 0 4 8 】

次に、本発明の第二の実施形態を説明する。

【 0 0 4 9 】

第一の実施形態の場合、実領域 1 3 2 が確保された後、その実領域 1 3 2 に格納されたデータが使用されなくなる場合について考慮されていない。そこで、第二の実施形態では、第一の実施形態に加え、使用されなくなった実領域 1 3 2 の

仮想ボリューム 1 0 0 への割当てを止める（以下「解放する」）ことを考慮する。

【 0 0 5 0 】

本実施形態においては、第一の実施形態の構成に加え、制御部 2 1 0 のメモリには、デフラグ処理プログラム 2 1 4 が格納されている。制御部 2 1 0 は、システム管理者のデフラグ処理開始指示を管理コンソール 1 4 を通して受け取り、デフラグ処理プログラム 2 1 4 を実行して、仮想ボリューム 1 0 0 に格納されたファイル等のデータを再配置する。具体的には、制御部 2 1 0 は、ストレージ装置 1 3 に格納されたファイルシステムの管理情報を読み取って、それに基づいて制御部 2 1 0 が実領域 1 3 2 のデータを空きの領域にコピーし、管理情報を書き換えることにより配置を変更する。なお、この再配置処理によるデータ破壊を防止するために、本実施形態ではシステム管理者は該仮想ボリューム 1 0 0 を使用するファイルシステムを、デフラグ処理開始指示の前にアンマウントしておく必要がある。

【 0 0 5 1 】

その後、制御部 2 1 0 は、仮想ボリューム 1 0 0 の記憶領域のうち、実領域 1 3 2 の割当てが不要になった記憶領域のアドレス範囲を特定し、特定した実領域 1 3 2 を解放し、仮想ボリューム管理テーブル 2 2 1 の対応するエントリ 3 1 7 を更新し、アクセス変換テーブル 2 2 4 の対応するエントリ 3 3 1 を更新する。

【 0 0 5 2 】

図 4 は、第二の実施形態における、実領域 1 3 2 を特定する処理の概念を示す図である。

【 0 0 5 3 】

図 4 は仮想ボリューム 1 0 0 におけるファイルの配置及び仮想ボリューム 1 0 0 と実領域 1 3 2 との対応関係を例示しており、図 4 （１）は制御部 2 1 0 のデフラグ処理プログラム 2 1 4 の実行前を、図 4 （２）はデフラグ処理プログラム 2 1 4 の実行後を示す。

【 0 0 5 4 】

図 4 （１）において、仮想ボリューム 1 0 0 の図示された範囲の記憶領域には

3個のファイル501a、501b及び501cが存在し、これらのファイルは、仮想ボリューム100の記憶領域中、501a1、501a2、501b及び501cの矩形の領域を占有している。一方、仮想ボリューム100の図示された範囲の記憶領域には、2個の実領域132が対応付けられている。ここで、制御部210は、デフラグ処理プログラム214を実行して、各ファイルを仮想ボリューム100におけるアドレス順に連続するように再配置する。

【0055】

その結果、再配置処理後は、図4(2)のように、左(左の方がアドレスが小さいと仮定する)から順に501a1、501a2、501b、及び501cに対応する記憶領域が連続して配置される。すると、501cの後ろにはまとまった空き領域が出現する(制御部210は各ファイルを再配置しているので、この空き領域の、仮想ボリューム100上のアドレス範囲を再配置後のファイルシステムの管理情報から知ることが出来る)。この空き領域には実領域132を割当てておく必要がない。即ち、図4(2)で、実領域132-2は解放することが出来る。

【0056】

制御部210は、デフラグ処理プログラム214の実行後、出現した空き領域の仮想ボリューム100上のアドレス範囲から対応する実領域132を仮想ボリューム管理テーブル221から検索し、見つかった実領域132を解放する。具体的には、制御部210は該当する実領域132の情報を仮想ボリューム管理テーブル221から削除し、実領域管理テーブル222の該当する実領域132に対応する仮想ボリューム識別子をnullにし、アクセス変換テーブル224に登録された実領域132の情報を削除する。

【0057】

次に、本発明の第三の実施形態を説明する。

【0058】

第一の実施形態では、仮想ボリューム100に割当てられる実領域132のサイズは固定であった。しかし、所定の処理、例えばフォーマット処理に基づくデータの書き込み要求では、比較的記憶容量の使用量が小さい処理が発生する。こ

の場合にも他の実施形態と同様に仮想ボリューム 1 0 0 に固定サイズの実領域 1 3 2 を割当てると、割当てた記憶領域に無駄が発生する。そこで、本実施形態では、処理内容に見合ったサイズの実領域 1 3 2 を仮想ボリューム 1 0 0 に割当てる。

【 0 0 5 9 】

本実施形態は第一の実施形態の構成に加え、制御部 2 1 0 のメモリにフォーマット処理プログラム 2 1 5 が格納されている。また、仮想ボリューム管理テーブル 2 2 1 のエントリ 3 1 7 に、対応する実領域 1 3 2 の記憶領域のサイズを示す情報が含まれる。また、実領域管理テーブル 2 2 2 のエントリ 3 2 5 に含まれる実領域 1 3 2 のサイズの情報は、本実施形態では第一の実施形態と異なり省略されない。エントリ 3 1 7 及びエントリ 3 2 5 にサイズの情報を含めるのは、上述したように、実領域 1 3 2 のサイズを可変長として、割当効率を上げるためである。

【 0 0 6 0 】

制御部 2 1 0 は、フォーマット処理プログラム 2 1 5 を実行することで、ホストプロセッサ 1 2 に代わり、仮想ボリューム 1 0 0 を使用するファイルシステムを初期化、即ちファイルシステム上のファイル及びディレクトリを全て消去し、新規にファイル及びディレクトリを作成可能な状態にする。このとき、仮想ボリューム 1 0 0 には、メタデータと呼ばれる、管理領域が書き込まれる。このとき書き込まれるメタデータ 1 個のサイズはそれほど大きくないが、仮想ボリューム 1 0 0 の記憶領域に一定の間隔に書き込まれるため、仮想ボリューム 1 0 0 のサイズが大きい場合は多数のメタデータが書き込まれ、割当効率が低下する。実領域 1 3 2 のサイズを可変長とするのは、この割当効率の低下を防ぐためである。

【 0 0 6 1 】

尚、フォーマット処理プログラム 2 1 5 は、管理者等の管理端末 1 4 を介した指示に基づき、あるいはホストプロセッサ 1 2 からの指示に基づき実行される。

【 0 0 6 2 】

図 5 は、第三の実施形態において、ホストプロセッサ 1 2 又は制御部 2 1 0 自身が生成した入出力要求を受け付けた際の制御部 2 1 0 の処理手順を示すフロー

チャートである。

【0063】

本処理手順には、第一の実施形態の図3で示した処理手順に加え、制御部210が受信した入出力要求がフォーマット処理に基づく入出力要求かどうかを判断する処理（ステップ2003）及びフォーマット処理である場合に、仮想ボリューム100に割当て実領域132のサイズを変更する処理（ステップ2004）が含まれる。尚、その他のステップの処理は第一の実施形態と同様であるので、ここでは説明しない。

【0064】

制御部210は、入出力要求で指定された仮想ボリューム100のアドレスに対応する実領域132がアクセス変換テーブル224に登録されていない場合、その入出力要求がフォーマット処理に基づく入出力要求か否かを判別する。具体的には、制御部210は、入出力要求がフォーマット処理プログラム215の実行により生成された場合はフォーマット処理と判断し、それ以外、即ちホストプロセッサ12から受信した場合はフォーマット処理でないと判断する（ステップ2003）。

【0065】

その後、制御部210は、仮想ボリューム100に割当て実領域132のサイズを他のデータ書き込み処理の際に割当てられる実領域132のサイズの1/32と指定する。この除数はシステム管理者が任意に定めることができる。さらに、制御部210はステップ2006と同様に実領域管理テーブル222から未使用の実領域132を検索する。その際、指定した大きさの空き実領域132がない場合は、それより大きな実領域132を2個に分割し、指定した大きさの空き実領域132を使用する。さらに制御部210は実領域132を検索した後、その実領域132を割当て、アクセス変換テーブル224、実領域管理テーブル222及び仮想ボリューム管理テーブル221を更新する。

【0066】

図6は、第三の実施形態における、フォーマット処理の概要を示す図である。

【0067】

図 6 は仮想ボリューム 1 0 0 のメタデータの配置及び実領域 1 3 2 との対応を示しており、図 6 (1) は実領域 1 3 2 のサイズが固定されていると仮定した場合、図 6 (2) は実領域 1 3 2 のサイズが可変である場合を示す。図 6 に示した仮想ボリューム 1 0 0 の記憶領域には 2 個のメタデータ 5 0 2 が書き込まれており、このメタデータの書き込みのためだけに、図 6 (1) では、2 つの実領域 1 3 2 (1 3 2 - 1 、 1 3 2 - 2) が割当てられている。それに対し、図 6 (2) では、メタデータ 5 0 2 のサイズを考慮した実領域 1 3 2 - 3 及び実領域 1 3 2 - 4 が割当てられる。

【 0 0 6 8 】

このように、ステップ 2 0 0 4 で実領域 1 3 2 のサイズを縮小することで、割当てた実領域 1 3 2 中で実際には使用されていない領域を小さくすることができる。

【 0 0 6 9 】

次に、本発明の第四の実施形態を説明する。

【 0 0 7 0 】

本実施形態では、第二の実施形態と同様に、制御部 2 1 0 は、一度仮想ボリューム 1 0 0 に割当てられた実領域 1 3 2 のうち、解放可能な実領域 1 3 2 を検索し解放する。ただし、実領域の解放のために使用される情報が、本実施形態ではジャーナルファイルシステムがホストプロセッサ 1 2 で使用される場合のログ情報である点が第二の実施形態とは異なる。

【 0 0 7 1 】

本実施形態では、第一の実施形態の構成に加え、制御部 2 1 0 のメモリにジャーナル解析プログラム 2 1 6 が格納され、仮想ボリューム管理テーブル 2 2 1 に、ジャーナル領域のアドレス範囲を登録するエントリ 3 1 8 及びメタデータの複製を登録するエントリ 3 1 9 とが含まれる。

【 0 0 7 2 】

本実施形態においては、制御部 2 1 0 はジャーナル解析プログラム 2 1 6 を実行して、仮想ボリューム 1 0 0 に配置されたジャーナル領域に書き込まれた情報を解析し、解放可能な実領域 1 3 2 がある場合に実領域 1 3 2 の解放処理を行う

【 0 0 7 3 】

ジャーナル領域のアドレス範囲が登録されるエントリ 3 1 8 は、入出力要求がジャーナル領域への書き込みか否かを判定するために制御部 2 1 0 によって参照される。仮想ボリューム 1 0 0 がジャーナルファイルシステムとしてフォーマットされていない場合、ジャーナル領域のアドレス範囲エントリ 3 1 8 は空のままになる。仮想ボリューム 1 0 0 がジャーナルファイルシステムとして初期化されるときには、管理者が管理コンソール 1 4 を介して、またはジャーナルファイルシステムのフォーマットプログラムが該ジャーナルファイルシステム中のジャーナルが格納されているアドレス範囲をエントリ 3 1 8 に書き込む。メタデータの複製エントリ 3 1 8 は、制御部 2 1 0 が、仮想ボリューム 1 0 0 に格納されるメタデータの複製を保存する際に使用される。

【 0 0 7 4 】

図 7 は、第四の実施形態において、入出力要求を受け付けた制御部 2 1 0 の処理手順を示すフローチャートである。

【 0 0 7 5 】

本処理では、第一の実施形態の図 3 に示した処理に加え、制御部 2 1 0 が、入出力要求がジャーナル領域への書き込みか否かを判断する処理（ステップ 2 0 0 7）及び解放可能な実領域 1 3 2 の解放を行う処理（ステップ 2 0 0 8）が行われる。ステップ 2 0 0 7 において制御部 2 1 0 は、書き込み先のアドレスがジャーナル領域範囲 3 1 8 の中にある場合に、ジャーナル領域への書き込みと判断し、ジャーナル解析プログラム 2 1 6 の処理、即ちステップ 2 0 0 8 へ進む。ステップ 2 0 0 8 において制御部 2 1 0 は、ジャーナル領域に書き込まれた情報を解析する。ジャーナル領域に書き込まれる情報とは具体的には、メタデータ 3 1 9 の一部とその（メタデータ 3 1 9 内の）オフセットであるので、これらを元に制御部 2 1 0 はメタデータ 3 1 9 を構築する。このメタデータ 3 1 9 にはファイルシステム上のファイルとそれが占めるデータ領域のアドレスとの対応が格納されているので、このメタデータ 3 1 9 の変更内容により、制御部 2 1 0 は実領域 1 3 2 の割当が不要になった領域の仮想ボリューム上のアドレスを検索する。その

後、そのアドレスを元に制御部 210 は仮想ボリューム管理テーブル 221 を検索し、見つかった実領域 132 を解放し、仮想ボリューム管理テーブル 221、アクセス変換テーブル 224 及び実領域管理テーブル 222 を更新する。

【0076】

図 8 は、第四の実施形態における、実領域 132 の解放処理の概要を示す図である。

【0077】

図 8 は、仮想ボリューム 100 におけるジャーナル領域、メタデータ、ファイルの配置及び実領域 132 との対応を示している。図 8 (1) は、ジャーナルファイルシステム上のファイル 501 の縮小前を、図 8 (2) は縮小後を示している。このようにファイル 501 が縮小される場合、縮小されるファイル 501 を管理しているメタデータ 502 をホストプロセッサ 12 が更新するが、それに先立ち、処理内容をジャーナルログとして記録するために、ホストプロセッサ 12 はジャーナル領域のエントリ 503 に情報を書き込む。このジャーナル領域のエントリ 503 には、メタデータ 502 の内容が含まれており、この内容からファイル 501 が縮小され、図 8 (2) に示した領域を占有するようになることが分る。

【0078】

その結果、図 8 (2) の場合は、実領域 132-4 に対応する領域が空きになるため、実領域 132-4 は解放可能になる。なお、メタデータ 502 には全てのファイル 501 が仮想ボリューム 100 のどの記憶領域を占有しているかを示す情報が登録されているため、制御部 210 は、メタデータ 502 に登録された内容から、実領域 132 に対応する仮想ボリューム 100 の記憶領域が空きになったかどうかを判断する。さらに、制御部 210 は、ジャーナル領域のエントリ 503 へデータが書き込まれると、エントリ 503 から構築した最新のメタデータをエントリ 319 に保存し、その結果から仮想ボリューム管理テーブル 221 のエントリ 319 を更新して、実領域 132-4 を解放する。つまり、以上に述べた、ファイル 501 の縮小、ジャーナル領域のエントリ 503 の書き込み、エントリ 319 の更新、実領域 132-4 の解放はこの順で行われることになる。

【0079】

尚、ホストプロセッサ12によるメタデータ502の書き込みは、ジャーナル領域のエントリ503の書き込み以降に、制御部210の処理と非同期に行われる。そのために本実施形態ではメタデータ502を読み込まず、代わりにエントリ319を使用する。

【0080】

さらに、ファイル501の削除によって実領域132が解放可能になる場合も、ジャーナル領域にその削除についての内容が記録されるので、上述した処理と同様に実領域132を解放することができる。

【0081】

このように、制御部210は、仮想ボリューム100でデータが割当てられなくなった実領域132を解放できる。

【0082】

次に、本発明の第五の実施形態を説明する。

【0083】

本実施形態では、上述した第二、第三の実施形態において仮想化スイッチ11で実行されていたデフラグ処理プログラム214等の処理が専用の計算機にて行われる。

【0084】

本実施形態では、図1で示した第一の実施形態の構成に、仮想化スイッチ11に接続される計算機である専用サーバ15が追加される。専用サーバ15は、制御部210に代わり、デフラグ処理プログラム214およびフォーマット処理プログラム215の実行を行う。制御部210の処理能力やメモリ容量には限りがあり、仮想ボリューム100の数や対応するファイルシステムの種類には限界がある。そのため、専用サーバ15でデフラグ処理やフォーマット処理を代行する。

【0085】

したがって、第五の実施形態においては、仮想化スイッチ11の制御部210のメモリには、上述した実施形態で説明したデフラグ処理プログラム214やフ

フォーマット処理プログラム 215 は格納されておらず、その代わり、仮想化スイッチ 11 に接続される専用サーバ 15 と仮想化スイッチ 11 との間の遣り取りを制御部 210 が制御するためのプログラムであるサーバ連携プログラム 217 がメモリに格納されている。

【0086】

制御部 210 は、デフラグ処理を行うときに、デフラグ処理プログラム 214 を実行する代わりに、サーバ連携プログラム 217 を実行し、専用サーバ 15 にデフラグ処理開始要求を送信し、専用サーバ 15 から仮想ボリューム 100 上で空きになった記憶領域のリストを受信する。次に制御部 210 は、この空きになった記憶領域のリストから対応する実領域 132 を仮想ボリューム管理テーブル 221 から検索し、見つかった実領域 132 を解放する。

【0087】

制御部 210 は、フォーマット処理を行うときに、フォーマット処理プログラム 215 を実行する代わりにサーバ連携プログラム 217 を実行し、専用サーバ 15 にフォーマット処理開始要求を送信する。その後制御部 210 の処理は第三の実施形態と同様である。

【0088】

このように、本実施形態では、仮想ボリューム 100 の数や対応するファイルシステムの種類を増やすことが出来る。

【0089】

尚、上述の実施形態では、仮想化を実現する装置としてスイッチを例にあげて説明したが、それ以外の装置、例えば計算機でもルーターでも良い。

【0090】

【発明の効果】

本発明によれば、効率的に計算機に記憶領域を割当てることができる。

【図面の簡単な説明】

【図 1】

本発明を適用したシステムの全体構成図である。

【図 2】

仮想化スイッチ 1 1 の内部構成を示す図である。

【図 3】

制御部 2 1 0 の入出力要求の処理手順を示すフローチャートである。

【図 4】

第二の実施形態におけるデータの再配置の処理の概要を示す図である。

【図 5】

第三の実施形態における制御部 2 1 0 の入出力要求の処理手順を示すフローチャートである。

【図 6】

第三の実施形態におけるフォーマット処理の概要を示す図である。

【図 7】

第四の実施形態における制御部 2 1 0 の入出力要求の処理手順を示すフローチャートである。

【図 8】

第四の実施形態における、実領域 1 3 2 の解放処理の概要を示す図である。

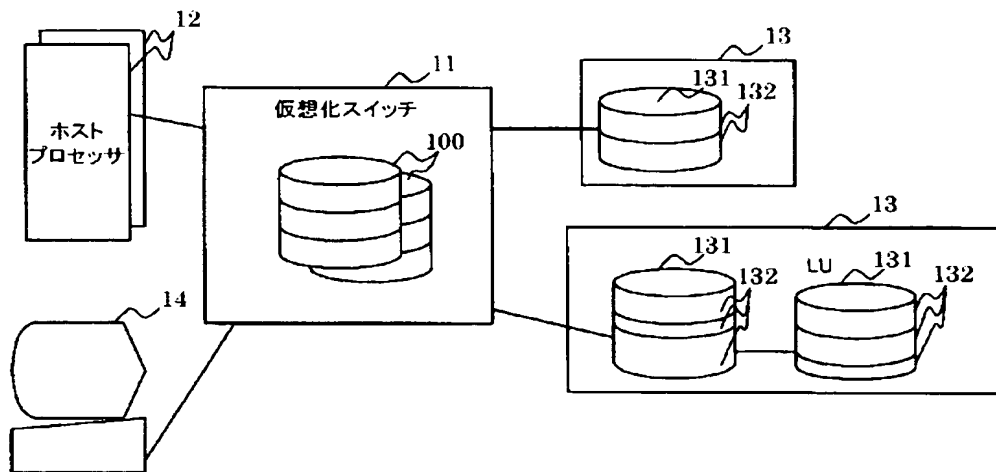
【符号の説明】

1 1 …仮想化スイッチ、1 2 …ホスト、1 3 …ストレージ装置、2 1 0 …制御部、2 2 0 …記憶部。

【書類名】 図面

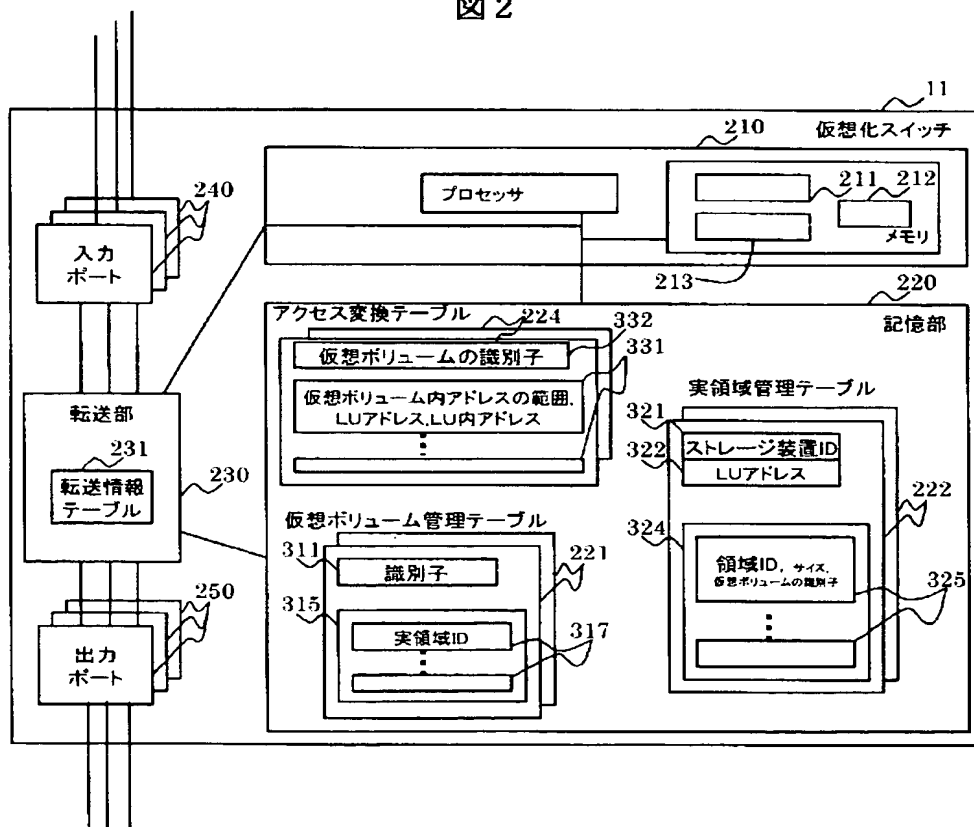
【図 1】

図 1



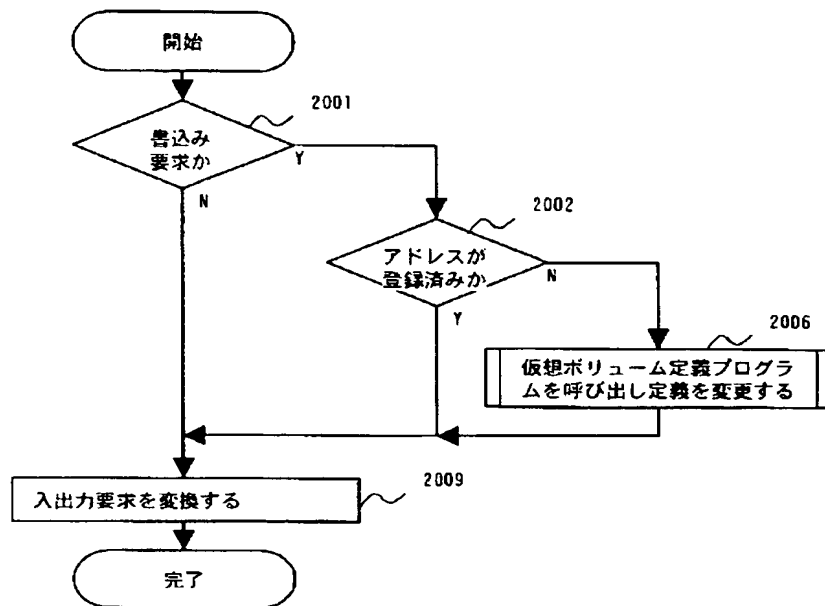
【図 2】

図 2



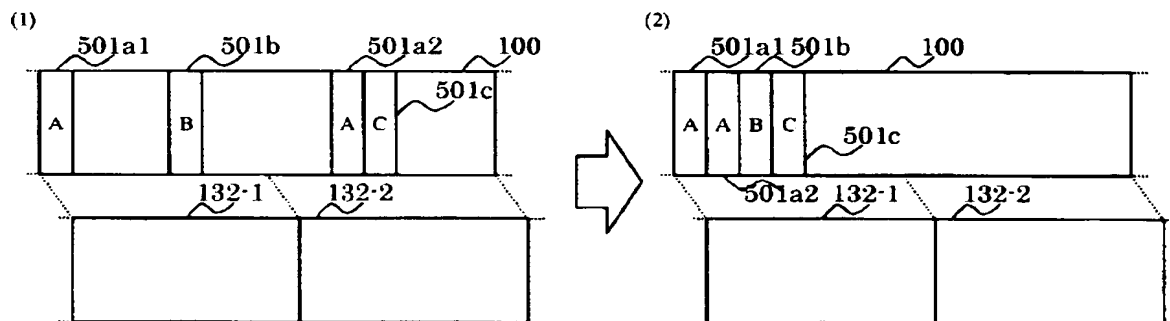
【図 3】

図 3



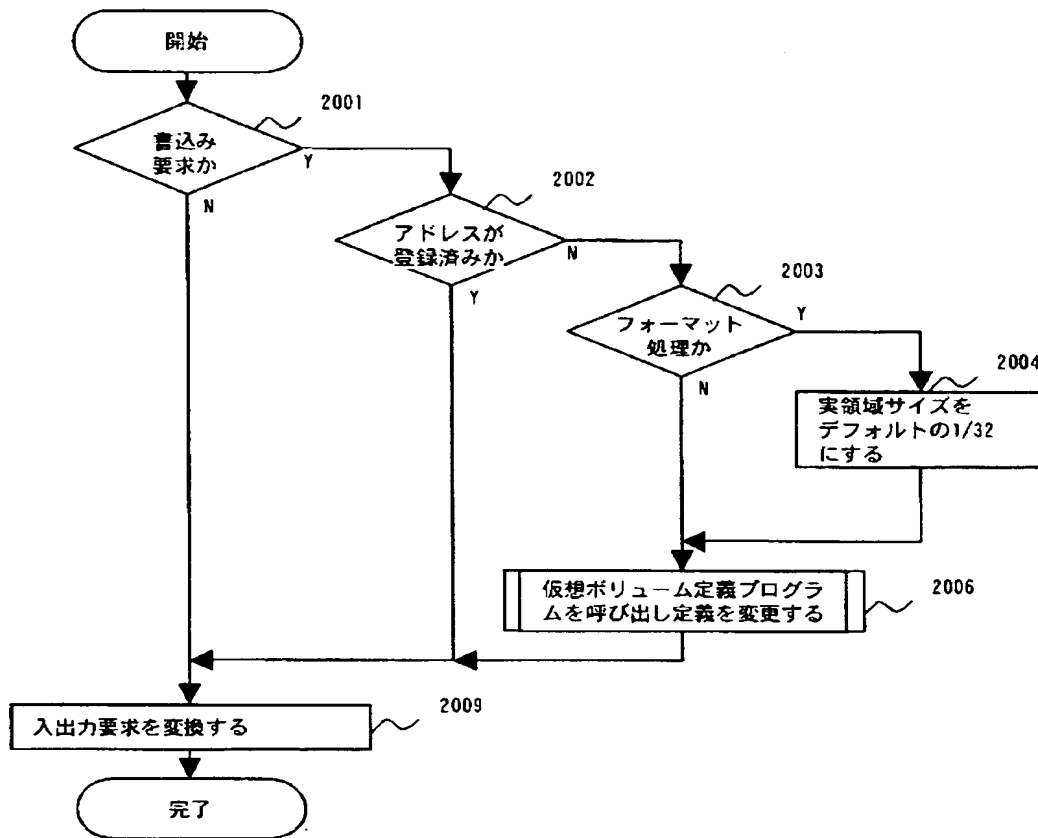
【図 4】

図 4



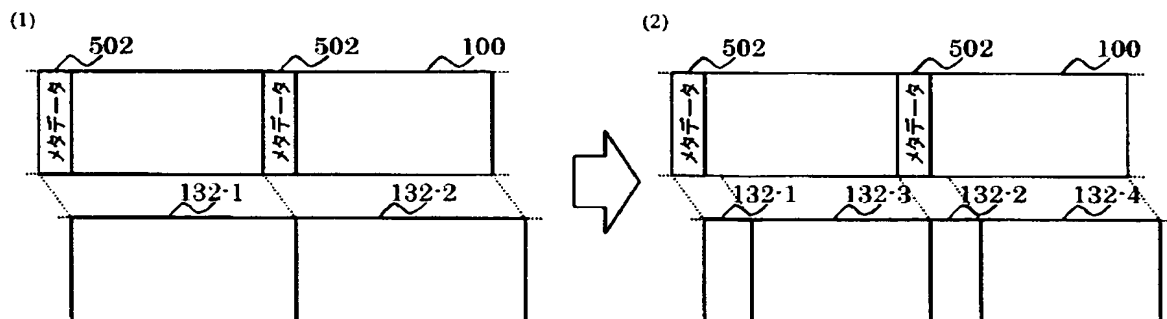
【図 5】

図 5



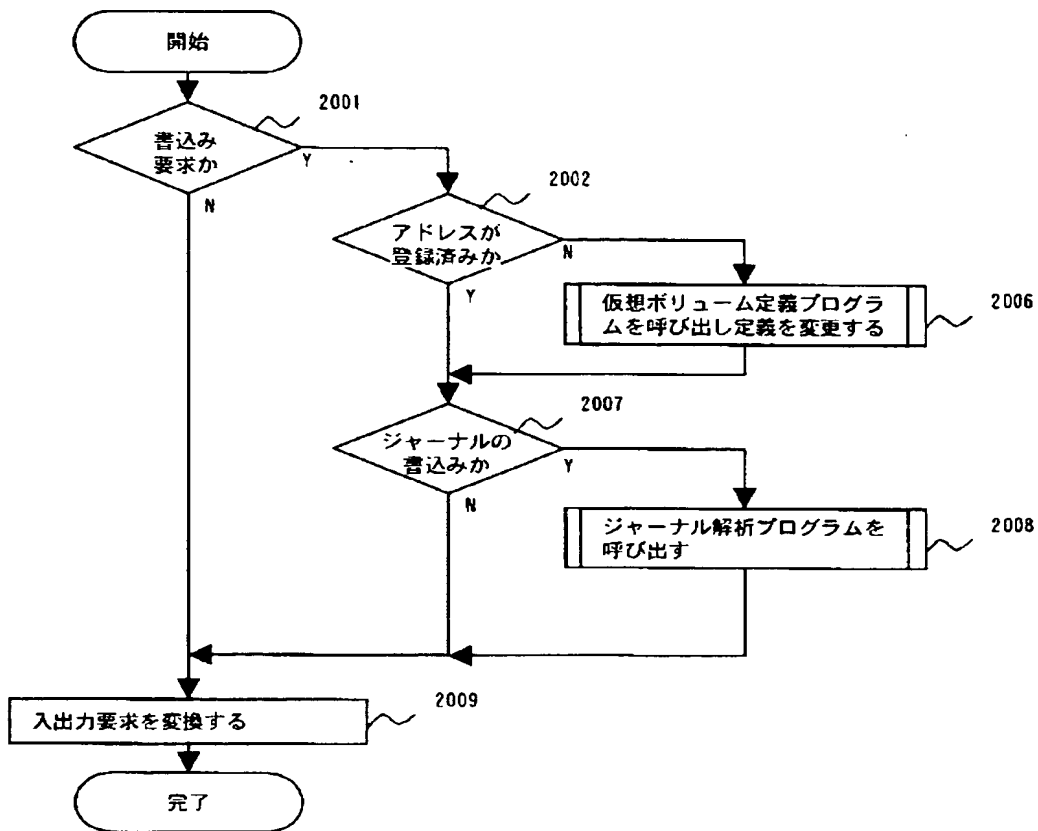
【図 6】

図 6



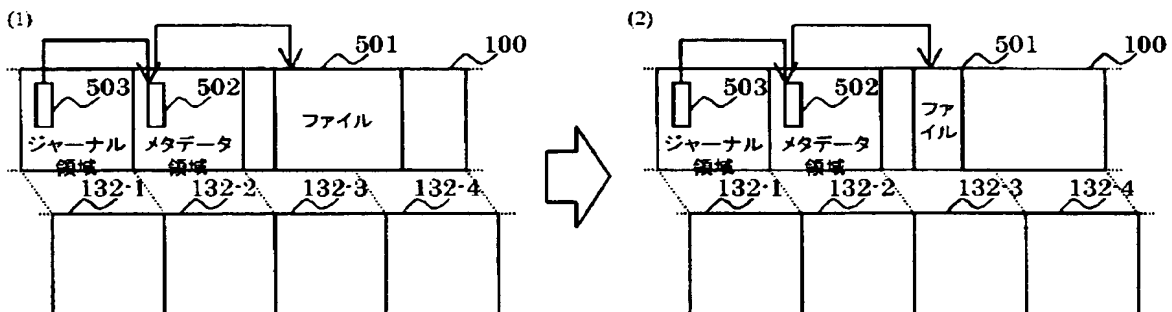
【図 7】

図 7



【図 8】

図 8



【書類名】 要約書

【要約】

【課題】

計算機に仮想化された記憶装置システムの記憶領域を割当てるときに、効率的に割当てることが出来ない。

【解決手段】

仮想化スイッチ 1 1 は、仮想ボリュームの作成時に、仮想ボリュームに実領域を割当てず、後にホストプロセッサ 1 2 が仮想ボリュームへ書き込みを行ったときに初めて、書き込みが行われた仮想ボリューム上の領域に実領域を割当てる。さらに、仮想化スイッチ 1 1 は、割当が不要となった実領域を解放する。また仮想化スイッチ 1 1 は、入出力要求の内容に応じて、割当て実領域の割当単位を小さくする。

【選択図】 図 2

認定・付加情報

特許出願の番号	特願 2 0 0 3 - 1 5 0 0 8 2
受付番号	5 0 3 0 0 8 8 0 3 8 6
書類名	特許願
担当官	第七担当上席 0 0 9 6
作成日	平成 1 5 年 5 月 2 9 日

< 認定情報・付加情報 >

【提出日】	平成15年 5月28日
-------	-------------

次頁無

特願 2 0 0 3 - 1 5 0 0 8 2

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 1 0 8]

1. 変更年月日

1 9 9 0 年 8 月 3 1 日

[変更理由]

新規登録

住 所

東京都千代田区神田駿河台 4 丁目 6 番地

氏 名

株式会社日立製作所